The Encoding Complexity of Network Coding with Two Simple Multicast Sessions

Wentu Song, Kai Cai, and Rongquan Feng

***Abstract*—The encoding complexity for network coding with one multicast session has been intensively studied from several aspects: e.g., the time complexity, the required number of encoding links and the required field size for a linear code solution. However, these issues are less understood for the network with multiple multicast sessions. Recently, C. C. Wang and N. B. Shroff declared that polynomial time can decide the solvability of the two simple multicast network coding (2-SMNC) problem. In this paper, we prove for the 2-SMNC networks: $1)$ the solvability can be determined with time $O(|E|)$; $2)$ a solution can be constructed with time $O(|E|)$; $3)$ an optimal solution can be obtained in polynomial time; $4)$ the number of encoding links required to achieve a solution is upper-bounded by $\max\{3, 2N-2\}$; and $5)$ the field size required to achieve a linear solution is upper-bounded by $\max\{2, \lfloor\sqrt{2N-7/4}+1/2\rfloor\}$, where $|E|$ is the number of links and N is the number of sinks of the underlying network. The bounds are shown to be tight and the algorithms to determine the solvability, to construct a solution and to construct an optimal solution are proposed.**

***Index Terms*—Network coding, encoding complexity, region decomposition.**

I. Introduction

A Communication network is described as a finite, directed, acyclic graph $G = (V, E)$, where a number of messages are generated at some nodes, named *sources*, and desired to receive by to some other nodes, named *sinks*. Messages are assumed to be independent random process with the elements taken from some fixed finite alphabet, usually a finite field. Network coding allows the intermediate nodes to "encode" the received messages before forwarding it, and has significant throughput advantages as opposed to the conventional store-and-forward scheme [1], [2]. The multicast network coding problem has been fully investigated and well understood by the network coding community. However, for the nonmulticast networks, the problem becomes even harder, and there were only a few results, for example, some deterministic results on the capacity region for some specific networks, such as single-source two-sink nonmulticast networks [3], directed cycles [6], degree-2 three-layer directed acyclic networks [8], and two simple multicast networks [4]. The outer bounds on the capacity region for general nonmulticast networks were obtained by information theoretic arguments [6]-[9] and the inner bounds were obtained by linear programming [10], [11]. In [12] it was proved that determining whether there exist linear network coding solutions for an arbitrary nonmulticast network is NP-hard.

Wentu Song and Rongquan Feng are with the School of Mathematical Sciences, Peking University, Beijing 100871, China (e-mails: songwentu@gmail.com; fengrq@math.pku.edu.cn).

Kai Cai is with Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China (e-mail: eecaikai@gmail.com).

An important issue for network coding problem is the encoding complexity, which has been intensively studied for multicast networks [14]-[19]. For nonmulticast networks, it remains challenging due to the intrinsic hardness of the nonmulticast network coding problem. In previous works the encoding complexity is generally studied from three aspects: the time complexity for constructing a network coding solution, the number of the required encoding nodes, and the required field size for achieving a network coding solution.

The time complexity is a fundamental issue. It is well known that a network coding solution can be achieved with polynomial time for multicast networks [14]. In [18], the authors first categorized the network links into two classes, i.e., the *forwarding links* and the *encoding links*. The forwarding links only forward the data received from its incoming links. While, the encoding links transmit *coded packets*, which need more resources due to the computing process and the equipping of encoding capabilities. It was shown that, in an acyclic multicast network, the number of encoding nodes (i.e., the tail of a encoding link) required to achieve the capacity of the network is independent of the size of the underlying network and bounded by h^3N^2, where N is the number of the sinks and h is the number of the source messages. The third aspect of encoding complexity is the required field size. As mentioned in [19], larger encoding field size may cause difficulties, i.e., either larger delays or larger bandwidths for the implementation of network coding, hence smaller alphabets are more preferred. For the multicast network, the required alphabet size to achieve a solution is upper bound by N, where N is the number of sinks (see [14]). In [17], the authors showed that an finite field with size $\sqrt{2N-7/4}+1/2$ is sufficient for achieving a solution of a multicast network with two source messages.

In this paper, we consider the encoding complexity for two simple multicast network coding problem (2-SMNC), where two unite rate messages are send from two sources and required by two sets of sinks respectively. If the two sink sets are identical, it is a multicast network coding problem, of which the solvability can be characterized by the well-known max-flow condition and its encoding complexity has been discussed as mentioned above. However, in the case the two sink sets are distinctly different, the situation becomes complicated. The recent work of C. C. Wang and N. B. Shroff [4] investigated this problem and showed that the solvability of the 2-SMNC problem can be characterized by *paths with controlled edge-overlap* condition under the assumption of

sufficient large encoding fields. They also proved that to decide the solvability of a 2-SMNC networks is polynomial time complexity and linear network codes are sufficient to achieve a solution. However, they did not consider the other issues of the encoding complexity.

This paper aims at the encoding complexity of 2-SMNC networks and the main contributions are listed as follows:

- We give algorithms to determine the solvability of the 2-SMNC problem with runtime $O(|E|)$; furthermore, we show that for a solvable network, a network coding solution can constructed with time $O(|E|)$, where $|E|$ is the number of the links of the underlying network;
- We give a polynomial time algorithm to construct the *optimal* solutions for the solvable 2-SMNC networks;
- We prove that the number of encoding links for achieving a solution is upper-bounded by $\max\{3, 2N-2\}$, where N is the number of sinks. Note that it is independent with the network size and only related to N. We also construct a instance to show this bound is tight.
- We prove that the required field size to achieve a linear solution is upper-bounded by $\max\{2, \lfloor\sqrt{2N-7/4}+1/2\rfloor\}$, which is *amazingly as the same small as the multicast case*. Also, this bound is shown to be tight by construction of a instance.
- To obtain these results, we proposed a *region decomposition* method, which promotes the *subtree decomposition* method for multicast networks [17]. Unlike subtree decomposition, we need not find a subgraph at first, also the regions are not necessarily being trees. Moreover, our method yields a unique region decomposition, namely the *basic region decomposition* for each network. Note that the subtree decomposition is in general not unique.

The technical line of this paper is as follows: Consider the original 2-SMNC network G. we first obtain its line graph $L(G)$, which can be regarded as a trivial region graph of G. Then we preform sequential *region contractions* on $L(G)$, and finally we obtain the basic region decomposition of G, namely, D^{**}. The solvability information of G then can be obtained from D^{**} by using the *region labeling operation*, and the network code solution can be obtained by assigning linear independent global encoding kernels to the regions of D^{**} using a decentralized manner. To give the optimal solution, we do further region contractions on D^{**} (in fact, this process can start with an arbitrary *feasible region graph*) and finally obtain a *minimal feasible region graph*. The global information such as the required encoding links and the required field size can be derived by analyzing the local structure of the minimal feasible region graph.

The rest of the paper is organized as follows. In Section II, we give the network model and some basic definitions. In Section III, we introduce the method of region decomposition. We give definitions of region, region decomposition, region graph, region contraction, codes on the region graph, feasible region graph, region labeling, and etc. We also derive some basic properties for these basic notions in this section. In Section IV, we decide the time complexity for solving the 2-SMNC problem by introducing the basic region decomposition D^{**}. We introduce the minimal feasible region graph and give the optimal solution in Sections V, the number of required encoding links is given in the same section. The required encoding field size is obtained in Section VI. Finally, we conclude the paper in Section VII.

II. Network Model and Notations

We consider the two simple multicast network coding problem (2-SMNC), of which the underlying network is assumed to be a finite, directed, acyclic graph $G=(V,E)$, where V is the set of nodes (vertices) and E is the set of links (edges). There are two sources $s_1, s_2 \in V$ and two sets of sinks $T_1=\{t_{1,1},\cdots,t_{1,N_1}\}, T_2=\{t_{2,1},\cdots,t_{2,N_2}\}\subseteq V$, where $s_i \notin T_i (i=1,2)$. Two messages X_1 and X_2 are generated at s_1 and s_2 and are demanded by T_1 and T_2 respectively. Note that $T_1 \neq T_2$ generally. The messages are assumed to be independent random variables taking values from a fixed finite field and a link $e=(u,v)$ is assumed of unit capacity, i.e., it can transmit one symbol from node u to v per transmission.

For $e=(u,v)\in E$, node u is call the tail of e and node v is called the head of e and denoted by $v=head(e)$ and $u=tail(e)$. For $e_1, e_2 \in E$, we call e_1 an *incoming link* of e_2 if $head(e_1)=tail(e_2)$. Denote by $In(e)$ the set of the incoming links of e. We assume $|In(e)|<M$ for each $e\in E$ and for some integer M.

We assume that for each source s_i there is one imaginary link from nowhere to s_i, called the X_i *source link*, and for each sink $t_{i,j}\in T_i$ there is one imaginary link from $t_{i,j}$ to nowhere, called the X_i *sink link*. The following terms are used in their self-evident meaning. An X_i *link* means the X_i source link or an X_i sink link. A *source* (resp. *sink*) *link* means the X_1 source (resp. sink) link or the X_2 source (resp. sink) link. Note that the source links have no tail and the sink links have no head, but this does not affect our discussion.

We assume that $In(e)\neq\emptyset$ if $e\in E$ is not a source link. Otherwise e has no impact on the network and can be removed from G.

Remark 2.1: Since $G=(V,E)$ is acyclic, E can be sequentially indexed as $e_1, e_2, e_3, \cdots, e_{|E|}$ such that 1) e_1 is the X_1 source link and e_2 is the X_2 source link; 2) $i<j$ if e_i is an incoming link of e_j. Note that such an index will be used in the sequel.

The network coding solutions of a 2-SMNC network are defined as follows.

Definition 2.2 (Network Coding Solution): A *network coding solution* (or *a solution* for short) of G over field $\mathbb{F}$ is a collection of functions $C=\{f_e:\mathbb{F}^2\to\mathbb{F}; e\in E\}$ such that

(1) If e is an X_i link $(i\in\{1,2\})$, then $f_e(X_1,X_2)=X_i$;
(2) If e is not a source link, then f_e can be computed from $f_{p_1},\cdots,f_{p_k}$, where $\{p_1,\cdots,p_k\}=In(e)$. This means that there is a $\mu_e:\mathbb{F}^k\to\mathbb{F}$ such that $f_e=\mu_e(f_{p_1},\cdots,f_{p_k})$.

The function f_e is called the *global encoding function* of e and μ_e is called the *local encoding function* of e. A solution C is called a *linear solution* if the global and local encoding functions are all linear functions over $\mathbb{F}$.

A network G is said to be *(linearly) solvable* if G has a (linear) solution over some finite field $\mathbb{F}$.

Remark 2.3: In the linear case, the global encoding function f_e of any $e \in E$ is in the form $f_e(X_1, X_2) = c_1 X_1 + c_2 X_2$, where $c_1, c_2 \in \mathbb{F}$. Hence f_e can be identified with the vector $d_e = (c_1, c_2) \in \mathbb{F}^2$ and C can be denoted by $C = \{d_e \in \mathbb{F}^2; e \in E\}$, where d_e is called the *global encoding kernel* of e. (1) and (2) of Definition 2.2 are equivalent to the following two conditions, respectively.

(1′) If e is an X_i link ($i \in \{1, 2\}$), then $d_e = \alpha_i$, where $\alpha_1 = (1, 0)$ and $\alpha_2 = (0, 1)$;

(2′) If e is not a source link, then d_e is an $\mathbb{F}$-linear combination of $\{d_p; p \in In(e)\}$.

Definition 2.4 (Forwarding Link and Encoding Link): Let $C = \{f_e : \mathbb{F}^2 \to \mathbb{F}; e \in E\}$ be a solution of G. e is called a *forwarding link* of C if $f_e = f_u$ for some $u \in In(e)$. Else, e is called an *encoding link* of C.

As in [13], we define the line graph of a network $G = (V, E)$, denoted by $L(G)$, as a directed, simple graph with vertex set E and edge set $\{(e_i, e_j) \in E^2; head(e_i) = tail(e_j)\}$. The line graph $L(G)$ is obviously finite and acyclic since G is finite and acyclic.

III. Region Decomposition

In this section, we introduce the region decomposition method for 2-SMNC networks.

A. Region and Region Graph

Definition 3.1 (Region and Region Decomposition): Let R be a non-empty subset of E and $e_0 \in R$. R is called a region of G generated by e_0 if any $e \in R \backslash \{e_0\}$ has an incoming link in R. Meanwhile, e_0 is called the *head* of R and is denoted by $e_0 = head(R)$. If E is partitioned into mutually disjoint regions $R_1, R_2, \cdots, R_K$, we say $D = \{R_1, R_2, \cdots, R_K\}$ is a region decomposition of G.

Let D be a region decomposition of G and $R \in D$. R is called the X_i *source region* if R contains the X_i source link; R is called an X_i *sink region* if R contains an X_i sink link, $i \in \{1, 2\}$. The X_1 source region and X_2 source region are called the *source region* and the X_1 sink region and X_2 sink region are called the *sink region*. For the sake of convenience, if R is not a source region, we call R a *non-source region*.

Obviously, there are many ways to obtain regions and decompose a network into *mutually disjoint* regions. For example, $\forall e \in E$, let $R_e = \{e\}$. Then R_e is a region and $D^* = \{R_e; e \in E\}$ is a region decomposition of G. We call D^* the *trivial region decomposition* of G.

We now show a nontrivial region decomposition which will also be used frequently in the sequel.

Example 3.2: Let G_1 be the network shown in Fig. 1(a) of which the line graph $L(G_1)$ is shown in Fig. 1(b). As illustrated in Fig. 2(a), $R_1 = \{e_1, e_3, e_4, e_{10}, e_{11}\}$, $R_2 = \{e_2, e_5, e_6\}$, $R_3 = \{e_7, e_8, e_9, e_{12}, e_{13}, e_{15}\}$, $R_4 = \{e_{14}, e_{16}, e_{18}\}$, $R_5 = \{e_{17}\}$, $R_6 = \{e_{19}\}$, $R_7 = \{e_{20}\}$, $R_8 = \{e_{21}\}$ are mutually disjoint regions of G_1 and $D = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8\}$ is a region decomposition of G_1, in which R_1 is the X_1 source region, R_2 is the X_2 source region, R_4 and R_8 are X_1 sink regions, and R_6 and R_7 are X_2 sink regions.

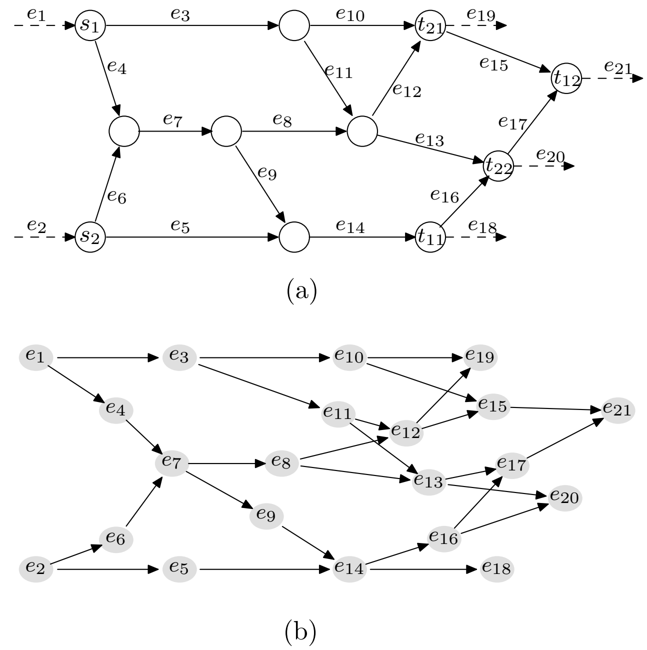


Fig 1. A 2-SMNC network G_1: (a) is the original network. The messages X_1 and X_2 are generated at source nodes s_1 and s_2 and are demanded by $T_1 = \{t_{11}, t_{12}\}$ and $T_2 = \{t_{21}, t_{22}\}$, respectively. The imaginary edges e_1 and e_2 are the X_1 source link and the X_2 source link respectively. Imaginary links e_{18}, e_{21} are X_1 sink links and imaginary links e_{19}, e_{20} are X_2 sink links. (b) is the line graph $L(G_1)$ of G_1.

We now define the region graph with respect to any region decomposition of G.

Definition 3.3 (Region Graph): Let D be a region decomposition of G. The region graph with respect to D, denoted by $RG(D)$, is a directed, simple graph with vertex set D and edge set $\{(R_i, R_j) \in D^2; In(head(R_j)) \cap R_i \neq \emptyset\}$, i.e., $(R_i, R_j) \in D^2$ is an edge of $RG(D)$ if and only if $head(R_j)$ has an incoming link in R_i.

If (R_i, R_j) is an edge of $RG(D)$, we call R_i a *parent* of R_j (R_j is called a *child* of R_i). Two regions R_i and R_j are said to be adjacent if R_i is a parent or a child of R_j. Denoted by $In(R_j)$ the set of all parents of R_j.

For network G_1 and its region decomposition D in Example 3.2, the corresponding region graph is depicted in Fig. 2(b).

Remark 3.4: By Definition 3.3, for the trivial region decomposition D^*, $RG(D^*)$ coincides with the line graph $L(G)$. For any region decomposition D, $RG(D)$ can be viewed as being *"contracted"* from $RG(D^*)$.

Lemma 3.5: Let D be a region decomposition of G, and $P, Q \in D$ such that P is a parent of Q. Then $P' = P \cup Q$ is a region of G with $head(P') = head(P)$ and $D' = D \cup \{P'\} \setminus \{P, Q\}$ is a region decomposition of G.

Proof: It can be easily verified by Definition 3.1. ∎

Definition 3.6 (Region Contraction): Under the conditions of Lemma 3.5, D' is called a contraction of D by combining P and Q. Correspondingly, the region graph $RG(D')$ is called a contraction of $RG(D)$ by combining P and Q.

Consider network G_1 and region decomposition D of Example 3.2. Clearly, $R_5 \cup R_8 = \{e_{17}, e_{21}\}$ is still a region of G_1 and $D' = \{R_1, R_2, R_3, R_4, R_5 \cup R_8, R_6, R_7\}$ is a contraction

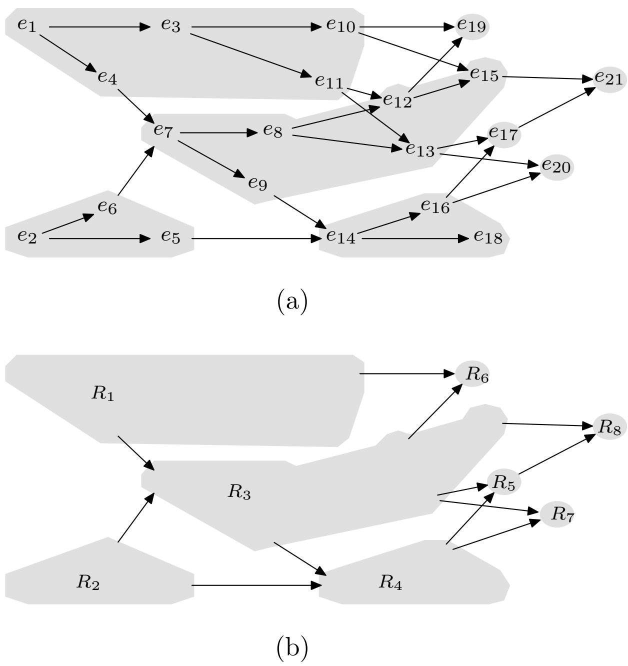


Fig 2. Region decomposition and region graph. (a) depicts the region decomposition $D = \{R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8\}$ of G_1, where G_1 is shown in Fig.1 and $R_1 = \{e_1, e_3, e_4, e_{10}, e_{11}\}$, $R_2 = \{e_2, e_5, e_6\}$, $R_3 = \{e_7, e_8, e_9, e_{12}, e_{13}, e_{15}\}$, $R_4 = \{e_{14}, e_{16}, e_{18}\}$, $R_5 = \{e_{17}\}$, $R_6 = \{e_{19}\}$, $R_7 = \{e_{20}\}$, $R_8 = \{e_{21}\}$; (b) is the corresponding region graph $RG(D)$.

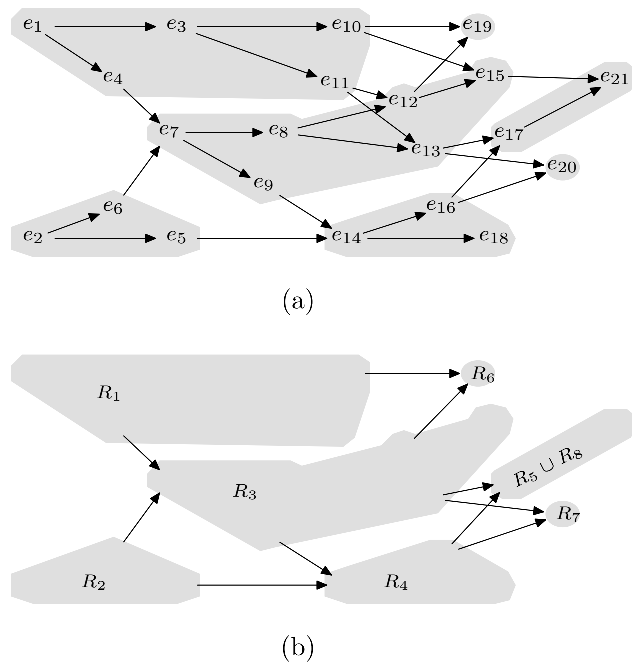


Fig 3. Region contraction. (a) depicts the region decomposition D' of G_1 and (b) is the region graph $RG(D')$, where G_1 and D are as in Example 3.2, and D' is the contraction of D by combining $R_5 = \{e_{17}\}$ and $R_8 = \{e_{21}\}$.

of D obtained by combining R_5 and R_8. D' and $RG(D')$ are illustrated in Fig. 3.

B. Codes on the Region Graph

Definition 3.7 (Codes on the Region Graph): Let D be a region decomposition of G and $\tilde{C} = \{f_R : \mathbb{F}^2 \to \mathbb{F}; R \in D\}$ be a collection of functions. $\tilde{C}$ is said to be a code of $RG(D)$ if the following two conditions hold.

(1) If R is an X_i source region or an X_i sink region ($i \in \{1, 2\}$), then $f_R(X_1, X_2) = X_i$;

(2) If R is a non-source region, then f_R is computable from $(f_{P_1}, \cdots, f_{P_k})$, where $\{P_1, \cdots, P_k\} = In(R)$. This means that there is a $\mu_R : \mathbb{F}^k \to \mathbb{F}$ such that $f_R = \mu_R(f_{P_1}, \cdots, f_{P_k})$.

Here, f_R is called the *global encoding function* of R and μ_R is called the *local encoding function* of R.

$RG(D)$ is said to be *feasible* if it has a code over some finite field $\mathbb{F}$. We also say a region decomposition D *feasible* if $RG(D)$ is feasible. A code $\tilde{C}$ is called a *linear code* if the global and local encoding functions are all linear functions.

Remark 3.8: Similar to the linear solution of G, the global encoding function f_R of a linear code can be identified with a vector $d_R = (c_1, c_2) \in \mathbb{F}^2$, called the *global encoding kernel* of R, and $\tilde{C}$ can be denoted by $\tilde{C} = \{d_R \in \mathbb{F}^2; R \in D\}$. Accordingly, (1) and (2) of Definition 3.7 are equivalent to the following conditions, respectively.

(1′) If R is an X_i source region or an X_i sink region ($i \in \{1, 2\}$), then $d_R = \alpha_i$, where $\alpha_1 = (1, 0)$ and $\alpha_2 = (0, 1)$;

(2′) If R is a non-source region, then d_R is an $\mathbb{F}$-linear combination of $\{d_P : P \in In(R)\}$.

Remark 3.9: A (linear) code of $RG(D^*)$ is exactly a (linear) solution of G, recall that $RG(D^*)$ is just the line graph of G. Thus, G is solvable if and only if $RG(D^*)$ is feasible.

The following lemma gives further observations on the relationship between the network coding solution and the codes on the region graph.

Lemma 3.10: Let D be a region decomposition of G. Then

(1) Let $C = \{f_e; e \in E\}$ be a (linear) solution of G such that $f_e = f_{head(R)}$ for any $R \in D$ and $e \in R$. Then $\tilde{C} = \{f_R; f_R = f_{head(R)}, R \in D\}$ is a (linear) code of $RG(D)$.

(2) Let $\tilde{C} = \{f_R; R \in D\}$ is a (linear) code of $RG(D)$, and let $C = \{f_e; e \in E\}$ such that $f_e = f_R$ for any $R \in D$ and $e \in R$. Then C is a (linear) solution of G.

Remark 3.11: In the above construction of C, we assign a same encoding kernel to a region, thus $e \in E$ is an encoding link of C only if e is the head of some non-source region.

Consider again network G_1 and region decomposition D shown in Fig. 2(a). A solution C of G_1 is depicted in Fig. 4(a) (by line graph $L(G_1)$). 4(b) shows the corresponding code $\tilde{C}$ of $RG(D)$.

The following results show that some kinds of the region contractions can maintain the feasibility of the region graphs.

Corollary 3.12: Suppose D is a feasible region decomposition and $\tilde{C} = \{f_R : \mathbb{F}^2 \to \mathbb{F}; R \in D\}$ is a code on $RG(D)$. Suppose $P, Q \in D$ are two adjacent regions and $f_Q = f_P$. Let $f_{P \cup Q} = f_P$ and D' be the contraction of D by combining P and Q. Let $\tilde{C}' = \{f_R; R \in D'\}$. Then $\tilde{C}'$ is a code of $RG(D')$ and thus $RG(D')$ is feasible.

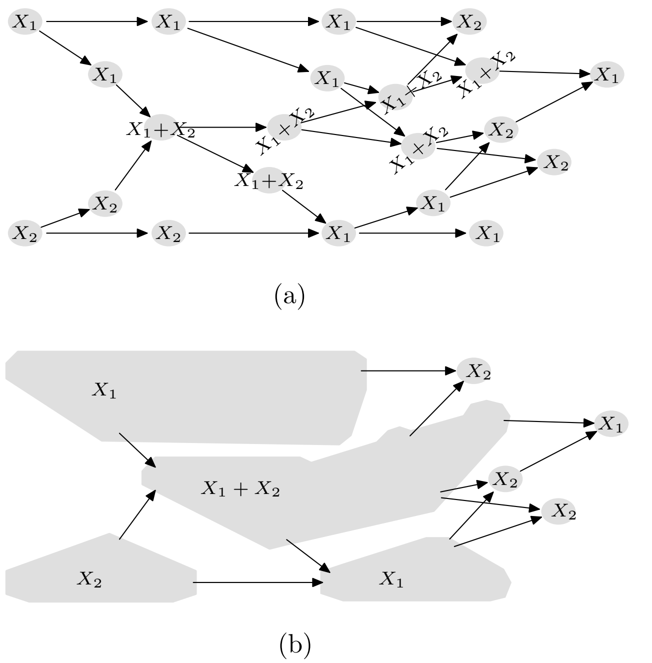


Fig 4. Network solution and the code on the region graph. Here, G_1 and D are as in Example 3.2. (a) depicts a solution of G_1; (b) is the corresponding code on $RG(D)$.

Reconsider G_1 and D in Example 3.2. Fig. 5(a) depicts a code of $RG(D)$ other than that in Fig. 4(b). Note that both R_5 and R_8 are assigned X_1, by Corollary 3.12, $D' = \{R_1, R_2, R_3, R_4, R_5 \cup R_8, R_6, R_7\}$ is feasible. For the same reason, $D'' = \{R_1, R_2, R_3, R_4 \cup R_5 \cup R_8, R_6, R_7\}$ is feasible. Codes of $RG(D')$ and $RG(D'')$ are depicted in Fig. 5 (b) and (c), respectively.

Lemma 3.13: Let D be a region decomposition of G and $P, Q \in D$ such that $In(Q) = \{P\}$, i.e., P is the unique parent of Q in $RG(D)$. Let D' be the contraction of D by combining P and Q. If $RG(D)$ is feasible then $RG(D')$ is feasible.

Proof: Let $\tilde{C} = \{f_R : \mathbb{F}^2 \to \mathbb{F}; R \in D\}$ be a code of $RG(D)$. Since $In(Q) = \{P\}$, we have $f_Q = \mu_Q(f_P)$, where $\mu_Q : \mathbb{F} \to \mathbb{F}$ is the local encoding function of R. If Q is an X_i sink region, we alter f_P by letting $f_P = X_i$ (By Definition 3.7, $f_Q = \mu_Q(f_P) = X_i$ is surjective. So μ_Q is surjective, hence is bijective since $\mathbb{F}$ is finite.). Otherwise, we alter f_Q by letting $f_Q = f_P$. It is easy to see that in both cases we obtain a code of $RG(D)$ such that $f_Q = f_P$. By Corollary 3.12, $RG(D')$ is feasible. ■

C. Feasibility and Region Labeling

In order to decide the solvability of G efficiently, we need further discussions on the feasibility of $RG(D)$. In the following, we first define two labeling operations.

Definition 3.14 (Region Labeling): Let D be a region decomposition of G. For $i \in \{1, 2\}$, the X_i labeling operation on $RG(D)$ is defined recursively as follows.

(1) A region which contains an X_i link is labeled with X_i;
(2) A region whose parents are all labeled with X_i is labeled with X_i.

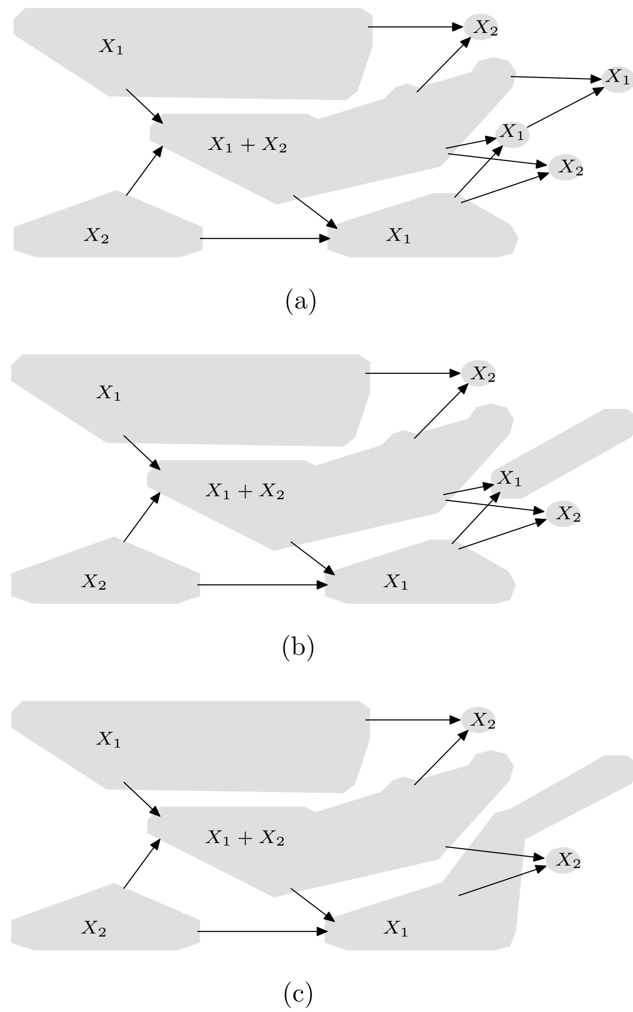
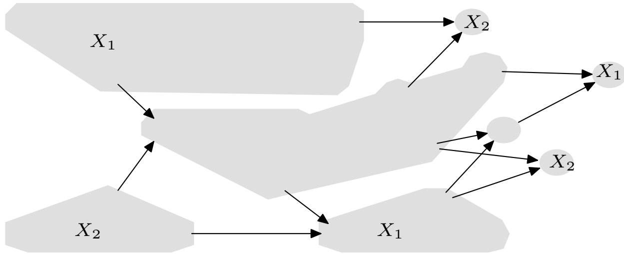


Fig 5. Codes on $RG(D)$, $RG(D')$ and $RG(D'')$ of G_1, where G_1 and D are as in Example 3.2, $D' = \{R_1, R_2, R_3, R_4, R_5 \cup R_8, R_6, R_7\}$ and $D'' = \{R_1, R_2, R_3, R_4 \cup R_5 \cup R_8, R_6, R_7\}$.

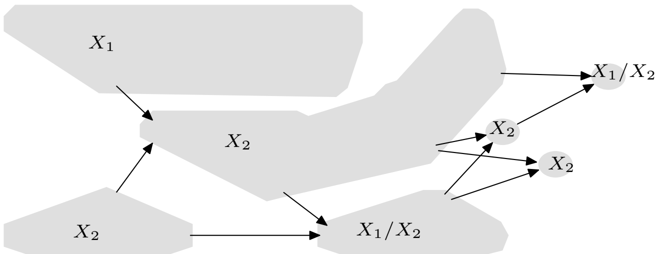
The X_i labeling operation is well defined because $RG(D)$ is acyclic. A region labeled with X_i is called an X_i *region*. A region which is neither X_1 region nor X_2 region is called a *coding region*. A region which is both X_1 region and X_2 region is called a *singular region*.

Consider network G_1 and region decomposition D in Example 3.2, the labeled region graph of $RG(D)$ is depicted in Fig. 6(a). Regions R_1, R_4 and R_8 are labeled with X_1 since R_1 contains X_1 source link e_1 and R_4 and R_8 contain X_1 sink links e_{18} and e_{21} respectively. Regions R_2, R_6 and R_7 are labeled with X_2 since R_2 contains X_2 source link e_2 and R_6 and R_7 contain X_2 sink links e_{19} and e_{20} respectively.

Now, let $D''' = \{R_1, R_2, R_3 \cup R_6, R_4, R_5, R_7, R_8\}$. By Lemma 3.5, it is a region decomposition of G_1. Similarly, R_1, R_4 and R_8 are labeled with X_1, and $R_2, R_3 \cup R_6$ and R_7 are labeled with X_2. Furthermore, R_4 is labeled with X_2 since the parents of R_4 are all labeled with X_2. Likewise, R_8 is labeled with X_2 since the parents of R_8 are all labeled with X_2. The labeled region graph $RG(D''')$ is depicted in Fig. 6(b). In this case, R_4 and R_8 are singular regions of $RG(D''')$.



(a)



(b)

Fig 6. Two examples of region labeling. (a) is the labeled region graph $RG(D)$ of G_1 and (b) is the labeled region graph $RG(D''')$, where G_1 and D are as in Example 3.2, and $D''' = \{R_1, R_2, R_3 \cup R_6, R_4, R_5, R_7, R_8\}$.

Form this example, we see that a same region may be labeled differently according to different region decompositions. In the following, we determine the feasibility of region decomposition through the labeled region graph. Firstly, we give some lemmas.

Lemma 3.15: Let D be a feasible region decomposition of G and $\tilde{C} = \{f_R : \mathbb{F}^2 \to \mathbb{F}; R \in D\}$ be a code of $RG(D)$. Then for any X_i region R, there exists a $\lambda_R : \mathbb{F} \to \mathbb{F}$ such that $f_R(X_1, X_2) = \lambda_R(X_i)$, that is, f_R depends only on X_i, $i = 1, 2$.

Proof: We prove this lemma by induction. Suppose the number of X_i regions of $RG(D)$ is K. Since $RG(D)$ is acyclic, all X_i regions of $RG(D)$ can be sequentially indexed as $R_1, R_2, \cdots, R_K$ such that $\ell < j$ if R_ℓ is a parent of R_j. By Definition 3.14, R_1 contains an X_i link. So by Definition 3.7, $f_{R_1}(X_1, X_2) = X_i$.

For $2 \leq k \leq K$, suppose f_{R_j} depends only on X_i for all $1 \leq j \leq k-1$. If R_k contains an X_i link, the result is evident. Else, by Definition 3.14, the parents of R_k are all X_i regions, hence $In(R_k) \subseteq \{R_1, \cdots, R_{k-1}\}$. By Definition 3.7, $f_{R_k} = \mu_{R_k}(f_{P_1}, \cdots, f_{P_r})$ depends only on X_i, where $\{P_1, \cdots, P_r\} = In(R_k)$ and $\mu_R : \mathbb{F}^r \to \mathbb{F}$ is the local encoding function of R_k. ■

Lemma 3.16: Suppose D is a feasible region decomposition of G. Then $RG(D)$ has no singular region.

Proof: Suppose $RG(D)$ has a singular region. Note that $RG(D)$ is acyclic, we can always find a singular region Q such that no parent of Q is a singular region. We declare that Q contains either an X_1 link or an X_2 link or both. (If Q contains neither X_1 links nor X_2 links, by Definition 3.14, all the parents of Q will be singular regions, which yields a contradiction.). Without loss of generality, we assume Q contains an X_1 link. Let $\tilde{C} = \{f_R : \mathbb{F}^2 \to \mathbb{F}; R \in D\}$ be a code of $RG(D)$. By Definition 3.7, $f_Q(X_1, X_2) = X_1$. On the other hand, by Lemma 3.15, $f_Q(X_1, X_2)$ depends only on X_2 since Q is also an X_2 region. A contradiction follows. ■

Lemma 3.17: Suppose D is a region decomposition of G such that $RG(D)$ has no singular region and each non-source region has at least two parents. Suppose $\tilde{C} = \{d_R \in \mathbb{F}^2; R \in D\}$ be a collection of vectors such that

(1) If R is an X_i region, $i \in \{1, 2\}$, then $d_R = \alpha_i$, where $\alpha_1 = (1, 0)$ and $\alpha_2 = (0, 1)$;
(2) If $R, Q \in D$ have a common child and are not both X_i regions for some $i \in \{1, 2\}$, then d_R and d_Q are linearly independent.

Then $\tilde{C}$ is a linear code of $RG(D)$.

Proof: Note that D contains no singular region, $\tilde{C}$ satisfies (1') of Remark 3.8. Now take a non-source region R, we only need to prove that d_R is an $\mathbb{F}$-linear combination of $\{d_P; P \in In(R)\}$. If the parents of R are all X_i regions for some $i \in \{1, 2\}$. By (1), $d_R = d_P = \alpha_i$ for all $P \in In(R)$. Otherwise, note that R has at least two parents, we can find two parents of R, say P_1 and P_2, such that d_{P_1} and d_{P_2} are linearly independent, and hence span $\mathbb{F}^2$. So d_R is an $\mathbb{F}$-linear combination of d_{P_1} and d_{P_2}. ■

Theorem 3.18: Let D be a region decomposition of G such that each non-source region in D has at least two parents. Then $RG(D)$ is feasible if and only if it has no singular region. Moreover, if $RG(D)$ is feasible, it has a linear code.

Proof: By Lemma 3.16, if $RG(D)$ is feasible, then $RG(D)$ has no singular region.

Conversely, if D contains no singular region, we can construct a linear code of $RG(D)$ as follows. Let $Q_1, \cdots, Q_J$ be the set of coding regions of $RG(D)$ and $\mathbb{F} = \{0, c_1 = 1, c_2, \cdots, c_{q-1}\}$ be a field of size $q \geq J + 1$. Let $\tilde{C} = \{d_R \in \mathbb{F}^2; R \in D\}$ such that

(1) If R is an X_i region, $i \in \{1, 2\}$, then $d_R = \alpha_i$;
(2) $d_{Q_j} = \beta_j$, where $\beta_j = (1, c_j), j = 1, \cdots, J$.

Note that $\alpha_1 = (1, 0), \alpha_2 = (0, 1)$ and $\beta_j = (1, c_j), j = 1, \cdots, J$ are mutually linearly independent. By Lemma 3.17, $\tilde{C}$ is a linear code of $RG(D)$. The result follows. ■

IV. Time Complexity for a Solution

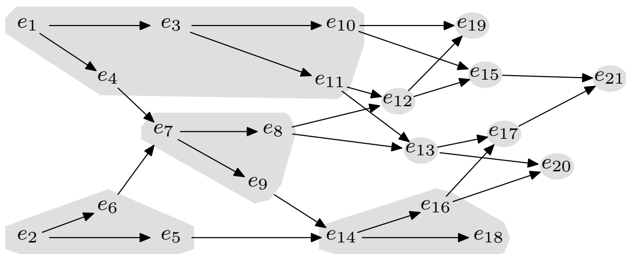
In this section, we give $O(|E|)$ time algorithms to determine solvability and to construct network coding solutions for 2-SMNC networks. By Theorem 3.18, if one can find out a region decomposition such that each non-source region has at least two parents, then the feasibility of the region graph can be inferred by searching the singular regions. In the following, we will show that for each 2-SMNC network, such a region decomposition exists. We first introduce a definition.

Definition 4.1 (Basic Region Decomposition): We call a region decomposition D^{**} a basic region decomposition if the following two conditions hold.

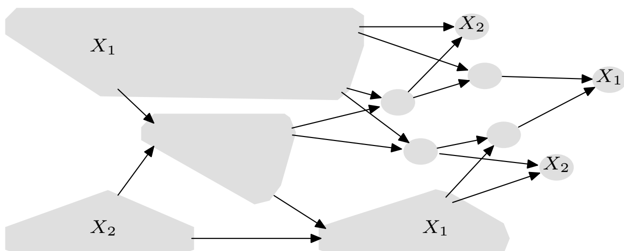
(1) For any region $R \in D^{**}$ and any link $e \in R \backslash \{head(R)\}$, $In(e) \subseteq R$;
(2) Each non-source region of D^{**} has at least two parents.

The following two examples demonstrate this notion.

Example 4.2: Consider the network G_1 in Example 3.2. See Fig. 7(a). Let $Q_1 = \{e_1, e_3, e_4, e_{10}, e_{11}\}$, $Q_2 = \{e_2, e_5, e_6\}$, $Q_3 = \{e_7, e_8, e_9\}$, $Q_4 = \{e_{12}\}$, $Q_5 = \{e_{13}\}$, $Q_6 = \{e_{14}, e_{16}, e_{18}\}$, $Q_7 = \{e_{15}\}$, $Q_8 = \{e_{17}\}$, $Q_9 = \{e_{19}\}$, $Q_{10} = \{e_{20}\}$, $Q_{11} = \{e_{21}\}$. It can be checked that $D^{**} = \{Q_1, \cdots, Q_{11}\}$ is a basic region decomposition of G_1.

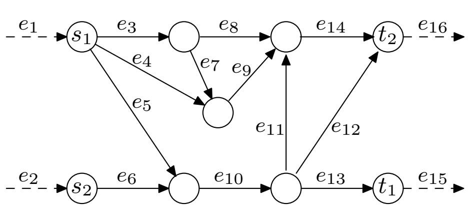


(a)

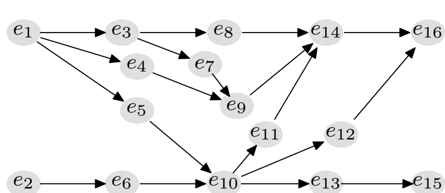


(b)

Fig 7. (a) depicts a basic region decomposition D^{**} of G_1 and (b) is the labeled region graph $RG(D^{**})$.



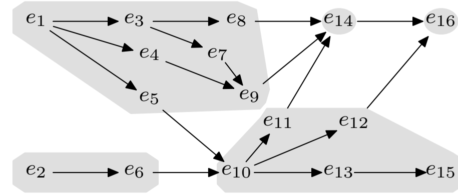
(a)



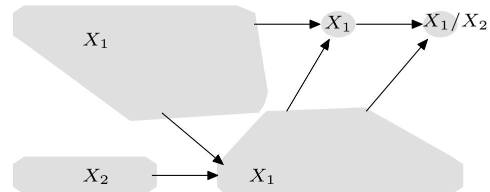
(b)

Fig 8. A 2-SMNC network G_2: (a) is the original network. The imaginary links e_1 and e_2 are the X_1 source link and X_2 source link respectively, and the imaginary links e_{15} and e_{16} are the X_1 sink link and X_2 sink link respectively. (b) is the line graph $L(G_2)$.

Example 4.3: Let G_2 be a 2-SMNC network shown in Fig. 8(a). The line graph $L(G_2)$ is shown in Fig. 8(b). As depicted in Fig. 9(a), let $R_1 = \{e_1, e_3, e_4, e_5, e_7, e_8, e_9\}$, $R_2 = \{e_2, e_6\}$, $R_3 = \{e_{10}, e_{11}, e_{12}, e_{13}, e_{15}\}$, $R_4 = \{e_{14}\}$, $R_5 = \{e_{16}\}$. It can be checked that $D^{**} = \{R_1, R_2, R_3, R_4, R_5\}$ is a basic region decomposition of G_2.



(a)



(b)

Fig 9. (a) depicts a basic region decomposition G_2 and (b) is the labeled region graph $RG(D^{**})$, where G_2 and D^{**} are as in Example 4.3.

Note that D, D', D'' and D''' (see Fig. 2, Fig. 3, Fig. 5, and Fig. 6, respectively) are not basic region decompositions of G_1 since they do not satisfy (1) of Definition 4.1. $L(G_1)$ is also not a basic region decomposition since it does not satisfy (2) of Definition 4.1.

In general, for an arbitrary 2-SMNC network G, we have the following result.

Theorem 4.4: G has a unique basic region decomposition.

Proof: Let E be indexed as in Remark 2.1. Consider Algorithm 1 in Fig. 10 with output $D^{**} = \{R_1, \cdots, R_K\}$. Clearly D^{**} satisfies the two conditions of Definition 4.1. Thus a basic region decomposition exists.

Now suppose $D = \{R_1, \cdots, R_K\}$ and $D' = \{Q_1, \cdots, Q_L\}$ are two basic region decompositions of G. We prove $D = D'$.

First, we prove that any $R_i \in D$ is contained in some region in D'. Let E be indexed as in Remark 2.1. Assume $R_i = \{e_{i_1}, e_{i_2}, \cdots, e_{i_n}\}$ such that $i_1 < i_2 < \cdots < i_n$. Without loss of generality, we assume $e_{i_1} \in Q_1$. We now prove $R_i \subseteq Q_1$. Otherwise, there exists an $e_{i_k} \in R_i$ such that $\{e_{i_1}, \cdots, e_{i_{k-1}}\} \subseteq Q_1$ and $e_{i_k} \in Q_j (j \neq 1)$. By (1) of Definition 4.1, $In(e_{i_k}) \subseteq \{e_{i_1}, \cdots, e_{i_{k-1}}\} \subseteq Q_1$. Note that $Q_1 \cap Q_j = \emptyset$ and by Definition 3.1, we can infer that $e_{i_k} = head(Q_j)$ and Q_1 is the only parent of Q_j, which contradicts to (2) of Definition 4.1.

Symmetrically, we can have $Q_1 \subseteq R_\ell$ for some $R_\ell \in D$. So $R_i \subseteq R_\ell$. Note that $R_i \cap R_\ell = \emptyset$ if $R_i \neq R_\ell$, we have $R_i = Q_1$. Note that R_i can be arbitrarily chosen from D, we have $D \subseteq D'$.

Similarly, we can have $D' \subseteq D$.

Thus $D' = D$ is the unique region decomposition of G. ■

In the following, we always use D^{**} to denote the the unique basic region decomposition of G. Now, we discuss

the solvability of G.

***Algorithm* 1**: *Region Decomposing* $(G = (V, E))$

$R_1 = \{e_1\}$;
$R_2 = \{e_2\}$;
$K = 2$;
$j = 3$;
While $j \leq |E|$ **do**
  if there is a $k \in \{1, \cdots, K\}$ such that $In(e_j) \subseteq R_k$ **then**
    $R_k = R_k \cup \{e_j\}$;
  else
    $K = K + 1$;
    $R_K = \{e_j\}$;
  end if
  $j = j + 1$;
end while
return $D^{**} = \{R_1, \cdots, R_K\}$;

Fig 10. The algorithm generates the basic region decomposition D^{**} of G. This algorithm is based on the assumption that E can be sequentially indexed as $e_1, e_2, e_3, \cdots, e_{|E|}$ such that 1) e_1 is the X_1 source link and e_2 is the X_2 source link; 2) $i < j$ if e_i is an incoming link of e_j.

Lemma 4.5: Let D^{**} be the basic region decomposition of G. Then

1) D^{**} can be obtained in time $O(|E|)$;
2) G is solvable if and only if $RG(D^{**})$ is feasible.

Proof: First, note that Algorithm 1 makes $|In(e_j)|$ times comparisons for each $e_j \in E, j \geq 3$. Thus, it can output D^{**} with time $O(|E|)$.

Second, according to Algorithm 1, D^{**} is in fact obtained from D^* by a series of region contractions, i.e., if the region $\{e_j\}$ has a unique parent R_k then combine e_j and R_k. Hence its feasibility remains unchanged (Lemma 3.13). Thus G is solvable if and only if $RG(D^*)$ is feasible (Remark 3.9) if and only if $RG(D^{**})$ is feasible. ■

Consider the basic region decomposition of G_1 in Example 4.2. The labeled region graph $RG(D^{**})$ is shown in Fig.7(b). One can see that D^{**} has no singular region, and hence is feasible (Theorem 3.18). By Lemma 4.5, G_1 is solvable.

Consider the basic region decomposition of G_2 in Example 4.2. The labeled region graph $RG(D^{**})$ is shown in Fig. 9(b). One can see that D^{**} has a singular region R_5 and hence is not feasible (Theorem 3.18). By Lemma 4.5, G_2 is not solvable.

Lemma 4.6: Let $D = \{R_1, \cdots, R_K\}$ be a region decomposition of G. The feasibility of $RG(D)$ can be decided in time $O(|E|)$. Moreover, if D is feasible, a linear solution of G can be constructed in time $O(|E|)$.

Proof: Let E be indexed as in Remark 2.1. The X_i labeling operation $(i \in \{1, 2\})$ on $RG(D)$ can be performed by Algorithm 2 in Fig. 11 and the feasibility of $RG(D)$ can be determined by Algorithm 3 in Fig. 12. The correctness of algorithm 3 is ensured by Theorem 3.18. Based on the proof of Theorem 3.18, a linear solution of G can be constructed by Algorithm 4 in Fig. 13. Clearly $|D| \leq |E|$, the runtime of these three algorithms are all $O(|E|)$. ■

***Algorithm* 2**: *X_i-Labeling* $(G = (V, E), RG(D))$

$j = 1$;
while $j \leq |E|$ **do**
  if $e_j \in R_k$ is an X_i link, **then**
    label R_k with X_i;
  end if
  $j = j + 1$;
end while
$k = 1$;
while $k \leq K$ **do**
  if the parents of R_k are all labeled with X_i, **then**
    label R_k with X_i;
  end if
  $k = k + 1$;
end while

Fig 11. The algorithm performs the X_i labeling operation $(i \in \{1, 2\})$ on the region graph $RG(D)$, where $D = \{R_1, \cdots, R_K\}$ is a region decomposition such that $i < j$ if R_i is a parent of R_j.

***Algorithm* 3**: *Determining feasibility* $(RG(D))$

$j = 1$;
while $j \leq K$ **do**
  if R_j is labeled with both X_1 an X_2, **then**
    return infeasible;
    stop;
  end if
end while
return feasible;

Fig 12. The algorithm determines the feasibility of $RG(D)$, where $D = \{R_1, \cdots, R_K\}$ have been labeled by X_1 labeling operation and X_2 labeling operation.

Now, we can conclude the section by the following theorem.

Theorem 4.7: Determining the solvability of G is an $O(|E|)$ time problem. Furthermore, if G is solvable, a linear solution of G can be constructed in time $O(|E|)$.

V. The Number of Encoding Links

Throughout this section, we assume that $G = (V, E)$ is a 2-SMNC network with two disjoint sink sets T_1 and T_2 [1], and

[1]If $t \in T_1 \cap T_2$, we can add two additional nodes t' and t'' and two additional links (t, t') and (t, t'') and replace t by t' in T_1 and t'' in T_2 respectively. Then any network coding solution for the old graph can be mapped bijectively to a network coding solution for the new graph without changing the encoding complexity.

***Algorithm* 4**: *Code Construction* $(RG(D))$

```
j = 1;
k = 1;
while j ≤ K do
  if R_j is labeled with X_i for an i ∈ {1,2} then
    f_{R_j} = X_i;
  else
    f_{R_j} = X_1 + c_k X_2;
    k = k + 1;
  end if
end while
j = 1;
while j ≤ |E| do
  if e_j ∈ R_k for a k ∈ {1,···,K} then
    f_{e_j} = f_{R_k};
  end if
end while
return C = {f_{e_j}; j = 1,···,|E|};
```

Fig 13. The algorithm constructs a linear solution of G. This algorithm is based on the proof of Theorem 3.18.

hence the number of sinks is equal to the number of sink links. We shall prove the following theorem.

Theorem 5.1: Let G be a solvable 2-SMNC network with N sinks, then G has a network coding solution with at most $\max\{3, 2N-2\}$ encoding links. There exist instances to achieve this bound.

To obtain this result, we need the concept of *minimal feasible region graph.*

Definition 5.2 (Minimal Feasible Region Graph): Let D be a feasible region decomposition of G. $RG(D)$ is said to be a minimal feasible region graph if the following two conditions hold.

(1) Deleting any link of $RG(D)$ results in a subgraph of $RG(D)$ which is not feasible.
(2) Combining any adjacent regions results in a contraction of $RG(D)$ which is not feasible.

We say that D is a *minimal feasible region decomposition* if $RG(D)$ is a minimal feasible region graph.

According to Definition 5.2, given a feasible region graph $RG(D)$ of G, if $RG(D)$ is not minimal feasible, one can always get a smaller feasible region graph, i.e., with less links and/or less nodes by deleting links and/or combining nodes of $RG(D)$. Once the deleting/combining process cannot be preformed, we get a minimal feasible region graph. By the manner of Lemma 3.10, we can obtain a network coding solution of G from a code of the minimal feasible region graph. The solution derived from the minimal feasible region graph will have less (or equal) encoding links than the solution derived from the original feasible region graph. From this sense, we call a solution constructed from the minimal feasible region graph as an *optimal solution* of G.

Consider the feasible region graphs $RG(D), RG(D')$ and $RG(D'')$ of G_1 in Fig. 5. It can be checked that $RG(D), RG(D')$ are not minimal feasible region graphs and $RG(D'')$ is minimal feasible. An optimal solution of G can be obtained from (c) of Fig. 5. The optimal solution of G_1 has only 4 encoding link, i.e., the head links of R_3, $R_4 \cup R_5 \cup R_8$, R_6, and R_7. In fact, the information of the required number of encoding links lies in the minimal feasible region graphs. To see this clearly, we first derive some properties of the minimal feasible region graph.

Theorem 5.3: Let D be a minimal feasible region decomposition of G. The following items hold.

1) Any non-source region has exactly two parents.
2) Two regions which are adjacent or have a common child cannot be both X_1 regions nor both X_2 regions.
3) Two adjacent coding regions have a common child.
4) If a coding region R is adjacent to an X_1 region (X_2 region), then there exists an X_1 region (X_2 region) P such that R and P have a common child.

Proof: 1) Let Q be a non-source region of G. Suppose Q has only one parent, namely, P. By Lemma 3.13, we can contract D by combining Q and P and obtain a new feasible region graph, which contracts to that D is minimal feasible. So Q has at least two parents.

Now, suppose Q has more than two parents. Let $\tilde{C} = \{d_R \in \mathbb{F}^2; R \in D\}$ be a code of $RG(D)$ constructed as in Theorem 3.18. There must be two parents of Q, say P_1 and P_2, such that d_Q is an $\mathbb{F}$-linear combination of d_{P_1} and d_{P_2}. Then delete the link(s) between Q and all the other parents, and we obtain a feasible subgraph with code $\tilde{C}$, which contradicts to that D is minimal feasible. Hence 1) holds.

2) Suppose P and Q are both X_1 regions (or both X_2 regions) and $\tilde{C}$ be a code of $RG(D)$ as in Theorem 3.18. Then $d_P = d_Q = \alpha_1$ (or $d_P = d_Q = \alpha_2$). If P and Q are adjacent, by Corollary 3.12, D can be contracted by combining Q and P without changing its feasibility. Similarly, if P and Q have a common child R, then deleting the link between Q and R results in a subgraph $RG(D)'$ of $RG(D)$ such that $\tilde{C}$ is still a code of $RG(D)'$. In both cases we derive contradictions and hence 2) holds.

3) Suppose $P, Q \in D$ are two adjacent coding regions which have no common child. Let $\tilde{C}$ be the code of $RG(D)$ as in Theorem 3.18. We alter $\tilde{C}$ by assigning the same global encoding kernel $d_P = d_Q$ to P and Q, and keep the rest of global encoding kernels unchanged. Since P and Q have no common child, this assignment is still a code of $RG(D)$ (Lemma 3.17). By Corollary 3.12, D can be contracted by combining P and Q without changing the feasibility, which is a contradiction and hence 3) holds.

4) Suppose R is adjacent to an X_1 region (X_2 region) Q and has no common child with any X_1 region (X_2 region). Let $\tilde{C}$ is the code of $RG(D)$ as in Theorem 3.18. We alter $\tilde{C}$ by letting $d_R = \alpha_1(\alpha_2)$, and keep the rest of global encoding kernels unchanged. Since R has no common child with any X_1 region (X_2 region), this assignment is still a code of $RG(D)$ (Lemma 3.17). By Corollary 3.12, D can be contracted by

combining R and Q without changing the feasibility, which is a contradiction and hence 4) holds. ■

For the sake of convenience, we say that a region Q is an *X_i-parent* (or an *X_i-child*) of a region R if Q is an X_i region and a parent (or a child) of R. The following corollary further shows some marvelous properties of the minimal feasible region graph.

Corollary 5.4: Let D be a minimal feasible region decomposition of G. The following items hold.

1) An X_i region is either an X_i source region or an X_i sink region $(i \in \{1, 2\})$.
2) A coding region has at lest two children which are sink regions.
3) If $R \in D$ is a coding region such that no child of R is a coding region, then R has two children, say R_1 and R_2, such that R_i is an X_i sink region and R_i has an X_j-parent, $i, j \in \{1, 2\}, j \neq i$.

Proof: 1) Let $R \in D$ be an X_i region. If R is neither an X_i source region nor an X_i sink region, i.e., R contains neither X_i source link nor X_i sink link, then the parents of R are all X_i regions (Definition 3.14), which contradicts to 2) of Theorem 5.3.

2) Let R be a coding region. Then by 1) of Theorem 5.3, R has two parents, say P_1 and P_2. By 2) of Theorem 5.3, they are neither both X_1 regions nor both X_2 regions. We distinguish the discussion into three cases.

Case 1: P_1 is a coding region and P_2 is an X_i region $(i \in \{1, 2\})$. First, consider P_1 and R. By 3) of theorem 5.3, P_1 and R have a common child Q_1. If Q_1 is an X_j region for some $j \in \{1, 2\}$, we halt. Else, if Q_1 is a coding region, then by 3) of theorem 5.3, R and Q_1 have a common child, say Q_2. Similarly, either Q_2 is an X_j region for some $j \in \{1, 2\}$ or R and Q_2 have a common child Q_3. Since $RG(D)$ is a finite graph, we can finally find an X_j-child Q_m of R. By 1), Q_m is a sink region.

Next, consider P_2 and R. Without loss of generality, we assume that P_2 is an X_1 region. By 4) of theorem 5.3, there exists an X_1 region P such that R and P have a common child W_1. If W_1 is an X_j region for some $j \in \{1, 2\}$, we halt. Else, if W_1 is a coding region, then by 3) of theorem 5.3, R and W_1 have a common child W_2. Similarly, either W_2 is an X_j region for some $j \in \{1, 2\}$ or R and W_2 have a common child W_3. Since $RG(D)$ is a finite graph, we can finally find an X_j-child W_n of R. By 1), W_n is a sink region. (Note that $W_n \neq Q_m$, which can be seen by sequently comparing the parents.)

Case 2: Both P_1 and P_2 are coding regions. Similar to case 1, we can find two child of R which are sink regions.

Case 3: P_1 is an X_1 region and P_2 is an X_2 region. Similar to case 1, we can find two child of R which are sink regions.

In all cases, we can find two child of R which are sink regions.

3) By 2), R has an X_i-child Q for some $i \in \{1, 2\}$. Without loss of generality, we assume that Q is an X_1 region. By 4) of Theorem 5.3, there is a region R_2 which is a common child of R and an X_1 region. By 2) of Theorem 5.3, R_2 is not an X_1 region. Note that no child of R is a coding region. So R_2 is an X_2 region.

Now consider R and R_2. By 4) of Theorem 5.3, there is a region R_1 which is a common child of R and an X_2 region. By 2) of Theorem 5.3, R_1 is not an X_2 region. Note that no child of R is a coding region. R_1 is an X_1 region. By 1), R_1 and R_2 are two sink regions meeting our requirements. ■

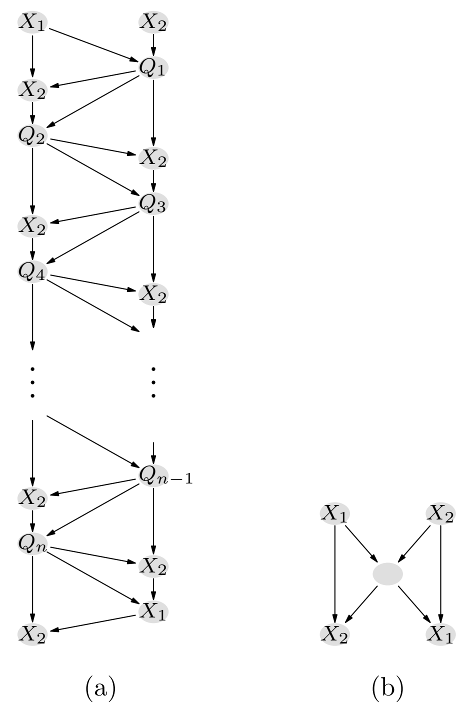


Fig 14. (a) is a minimal feasible region graph with $n \geq 2$ coding regions $\{Q_1, Q_2, \cdots, Q_n\}$ and $N = n + 2$ sink regions. For $j \in \{1, \cdots, n-1\}$, Q_j and Q_{j+1} have a common child of an X_2 sink region, and Q_{j+1} is a common child of Q_j and an X_2 sink region. (b) is a minimal feasible region graph with one coding region and 2 sink regions.

Theorem 5.5: Let D be a minimal feasible region decomposition of G with n coding regions. Then $n \leq \max\{1, N-2\}$, i.e., $n \leq 1$ when $N = 2$ and $n \leq N - 2$ when $N \geq 3$, where N is the number of sinks of G.

Proof: Let K be the number of sink regions of D. Obviously, $K \leq N$ since each sink region contains at least one sink link. Let J be the number of edges from a coding region to a sink region.

Suppose D has $n \geq 2$ coding regions, we prove $n \leq N - 2$ by counting J in two different ways. Firstly, note that $RG(D)$ is acyclic, we index D according to the upstream-to-downstream order, i.e., $\forall R, R' \in D$, $R < R'$ if R is a parent of R'. Let P and Q be the two coding regions with the biggest indexes and $P < Q$. We distinguish the following two cases to discuss.

Case1: Q is a child of P.

By 1) of theorem 5.3, the K sink regions of $RG(D)$ have exactly $2K$ parents (not necessarily different). We declare that including these $2K$ parents, at least 2 of them are not coding regions. (Noticing the index, no child of Q is a coding region. By 3) of Corollary 5.4, we can find two children R_1 and R_2 of Q, such that R_i is an X_i sink region and has an X_j-parent $(i \neq j)$.) Thus, we have $J \leq 2K - 2$.

On the other hand, by 2) of Corollary 5.4, except Q, the $n-1$ coding regions have at least $2(n-1)$ children which are sink regions. We declare Q has three children which are sink regions. (Note that P and Q are adjacent, by 3) of Theorem 5.3, P and Q have a common child, say R_3. Since P and Q are the coding regions with the biggest indexes, R_3 could not be a coding region. By 1) of Corollary 5.4, R_3 is a sink region. By comparing the parent set, we have $R_3 \notin \{R_1, R_2\}$.) Hence, we have $2(n-1)+3 \leq J$.

By the discussions above, we have $2n+1 \leq 2K-2$. Note that $K \leq N$ and n is an integer, we have $n \leq N-2$.

Case2: P and Q are not adjacent.

By 1) of theorem 5.3, the K sink regions of $RG(D)$ have exactly $2K$ parents (not necessarily different). We declare that including these $2K$ parents, at least 4 of them are not coding regions. (By the index, no child of P and/or Q is a coding region. By 3) of Corollary 5.4, P has two children, say R_1, R_2, such that R_i is a X_i sink region and each of them has an X_j-parent $(i \neq j)$. Similarly, Q also has two children W_1, W_2, such that W_i is an X_i sink region and each of them has an X_j-parent $(i \neq j)$.) Thus, we have $J \leq 2K-4$.

On the other hand, by 2) of Corollary 5.4, n coding regions have at least $2n$ children of sink regions. So $2n \leq J$. Thus, $2n \leq 2K-4$, and we have $n \leq N-2$.

By the above discussions, we see that if the network has 2 or more coding regions, it has at least 4 sinks (note that $N \geq n+2$). So, if $N = 2, 3$, $RG(D)$ has at most 1 coding region. The result follows. ■

Theorem 5.6: There exist instances which achieve the bound $n = \max\{1, N-2\}$ in Theorem 5.5.

Proof: Fig. 14 demonstrate the instances. Note that (a) has $n \geq 2$ coding regions and $N = n+2$ sink regions and it is feasible since it has no singular region. We can verify that this graph satisfies the two conditions of Definition 5.2. (b) shows the case of $N = 2$, a minimal feasible region graph $RG(D)$ with one coding region. ■

Now we can prove the main result of this section.

Proof of Theorem 5.1: Let D be a minimal feasible region decomposition of G. By Lemma 3.10, G has a network coding solution C such that $e \in E$ is an encoding link only if e is the head of a non-source region in D. By 1) of corollary 5.4, a non-source regions is either a coding region or a sink region. Note that the number of sink regions is at most N. By Theorem 5.5, when $N = 2$, $n \leq 1$ and when $N \geq 3$, $n \leq N-2$, we have that the number of encoding links is at most $\max\{3, 2N-2\}$.

Obviously, the networks with the minimal feasible region graphs of Fig. 14 can achieve this bound. ■

Algorithm 5 in Fig. 15 reduces the basic region graph $RG(D^{**})$ of G to a minimal feasible region graph $RG(D_m)$. The correctness of this algorithm is obvious. For each non-source region $R_j \in D^{**}$, Algorithm 5 makes $|In(head(R_j)|$ times verifications of (1) and (2) of Definition 5.2. Each time of the verification can be done by algorithm 2 and algorithm 3 with polynomial time. Note that $|In(R_j)| \leq |In(head(R_j))|$, we have that Algorithm 5 is also a polynomial time algorithm. After we get the minimal feasible region graph by Algorithm 5, an optimal solutions of 2-SMNC networks can be constructed by Algorithm 4 in polynomial time.

***Algorithm* 5**: *Minimal Region Graph* $(RG(D^{**}))$
  $D_m = D^{**}$ and $RG(D_m) = RG(D^{**})$;
  $j = 3$;
  while $j \leq |D^{**}|$ **do**
    for each $P \in In(R_j)$ **do**
      if combining R_j and P results in a feasible contraction D of D_m **then**;
        $D_m = D$ and $RG(D_m) = RG(D)$;
      else if Deleting the edge from P to R_j results in a feasible subgraph G **then**
        $RG(D_m) = G$;
      end if
    $j = j+1$;
  end while
  return $RG(D_m)$;

Fig 15. The algorithm reduces the basic region graph $RG(D^{**})$ to a minimal feasible region graph $RG(D_m)$. This algorithm is based on the fact that $D^{**} = \{R_1, \cdots, R_K\}$ can be sequently indexed such that 1) R_1 and R_2 are the two source regions; 2) $i < j$ if R_i is a parent of R_j.

VI. Bound of Field Size

In this section, following a same technical line as [17], we derive a tight upper bound on the required field size for the 2-SMNC problem. The result amazingly shows it is not necessary to use a larger field for 2-SMNC network than for one multicast session with two single rate flows[17].

Let $RG(D)$ be a minimal feasible region graph of a 2-SMNC network G having n coding regions $Q_1, \cdots, Q_n$. We first define the *associated graph* of $RG(D)$.

Definition 6.1: The associated graph of $RG(D)$, Ω_D is an undirected graph with $n+2$ vertices $X_1, X_2, Q_1, \cdots, Q_n$ and its edge set includes the following ones.

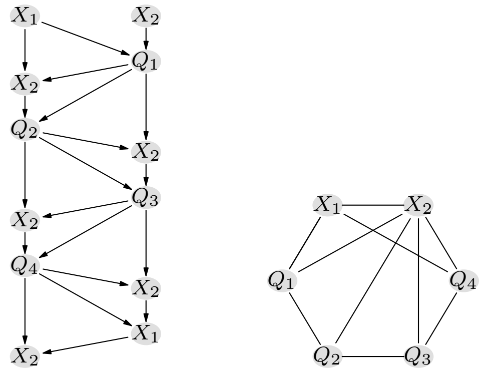
1) (X_1, X_2). It is called the red edge of Ω_D.
2) (Q_i, Q_j), if Q_i and Q_j have a common child. It is called a blue edge of Ω_D;
3) (Q_i, X_j), if Q_i have a common child with some X_j region $(j = 1, 2)$. It is called a green edge of Ω_D.

Example 6.2: For the minimal feasible region graph in Fig. 16(a), its associated graph Ω_D is shown in Fig. 16(b).

The problem of constructing a code of $RG(D)$ can be translated into a vertex coloring problem on Ω_D. First, we give a lemma.

Lemma 6.3: A field of size $q \geq \chi(\Omega_D) - 1$ is sufficient to achieve a linear solution of $RG(D)$, where $\chi(\Omega_D)$ is the chromatic number of Ω_D.

Proof: Let $F = \{0, c_1 = 1, c_2, \cdots, c_{q-1}\}$ be a field with size $q \geq \chi(\Omega_D) - 1$. Let $\alpha_1 = (1, 0), \alpha_2 = (0, 1)$ and $\beta_j = (1, c_j), j = 1, \cdots, q-1$. Let $\rho : V(\Omega_D) \to \{\alpha_1, \alpha_2, \beta_1, \cdots, \beta_{q-1}\}$ be a $q+1$-coloring of Ω_D such that $\rho(X_i) = \alpha_i$ for $i \in \{1, 2\}$. Let $\tilde{C} = \{d_R = \rho(R); R \in D\}$.



(a) (b)

Fig 16. (a) is a minimal feasible region graph; (b) is the associated graph.

It is easy to check $\tilde{C}$ satisfies the conditions of Lemma 3.17, hence is a linear code of $RG(D)$. ∎

Lemma 6.4: Let D be a minimal feasible region decomposition of G with $n \geq 1$ coding regions. The X_1 source region and X_2 source region have a common child.

Proof: Since $RG(D)$ is acyclic, its regions can be ordered in an upstream-to-downstream fashion, i.e., $\forall R, R' \in D$, $R < R'$ if R is a parent of R'. By 1) of Theorem 5.3, non-source region in D has exactly two parents in $RG(D)$. This implies that the third region must be a common child of the X_1 source region and the X_2 source region. ∎

Lemma 6.5: Let D be a minimal feasible region decomposition of G. Every vertex in Ω_D has degree at least 2.

Proof: 1) Vertex X_i, $i \in \{1, 2\}$: Since $RG(D)$ is finite and acyclic, there must be a coding region R such that no child of R is a coding region. By 3) of Corollary 5.4, both (R, X_1) and (R, X_2) are edges of Ω_D. Moreover, (X_1, X_2) is an edge of Ω_D.

2) Coding regions: Suppose R is a coding region. We have the following two cases.

Case 1: No child of R is coding region. As proved in 1), R has degree at least 2.

Case 2: R has a child Q which is a coding region. By 3) of Theorem 5.3, R and Q have a common child. Hence (R, Q) is an edge of Ω_D. Moreover, by 2) of Corollary 5.4, R has a child W which is an X_i sink region for some $i \in \{1, 2\}$. By 4) of Theorem 5.3, R has a common child with an X_i region. Hence (R, X_i) is an edge of Ω_D.

By all the discussions above, each vertex of Ω_D has degree at least 2. ∎

Lemma 6.6: [[21], Ch. 9] Every k-chromatic graph has at least k vertices of degree at least $k-1$. There exist configurations for which it is necessary.

Theorem 6.7: Suppose G be a solvable 2-SMNC network with N sinks. Then G has a linear solution over the field with size no larger than $\max\{2, \lfloor\sqrt{2N - 7/4} + 1/2\rfloor\}$.

Proof: If $N = 2$, Then a binary field is sufficient for a solution, as has been proved in [4]. We prove that the field of size $\lfloor\sqrt{2N - 7/4} + 1/2\rfloor$ is sufficient for a solution when $N \geq 3$.

Let D be a minimal feasible region decomposition of G with n coding regions and K sink regions. Let J be the number of edges of the associated graph, Ω_D and let $k = \chi(\Omega_D)$ be the chromatic number of Ω_D. We count J in two different ways.

By Lemma 6.5 and 6.6, each vertex of Ω_D has degree at least 2 and at least k vertices of degree at least $k-1$. We have $[k(k-1) + 2(n+2-k)]/2 \leq J$.

On the other hand, by 1) of Theorem 5.3, a region is a common child of two regions if and only if it is a non-source region. By 1) of Corollary 5.4, a non-source region is either a coding region or a sink region. So there are $n + K$ regions which are common children of two regions. Thus, $J \leq n+K$.

Combining the two inequalities, we have:

$$[k(k-1) + 2(n+2-k)]/2 \leq n + K.$$

Noting that $K \leq N$, and solving the inequality, we have,

$$k \leq \sqrt{2N - 7/4} + 3/2.$$

By Lemma 6.3, a field with size no larger than $\sqrt{2N - 7/4} + 1/2$ is sufficient for a linear solution. ∎

In the following, we show the tightness of the bound. To do this, we first give a lemma.

Lemma 6.8: Let $\mathbb{F}$ be a field of size q and $\gamma_1, \cdots, \gamma_k$ are k vectors in $\mathbb{F}^2$ such that any two of them are linearly independent. Then $k \leq q + 1$.

Proof: Suppose $k \geq q + 2$. Suppose $\gamma_i = (a_i, b_i), i = 1, \cdots, k$. We have the following two cases.

Case 1: There exist $i_1 \neq i_2$ such that $a_{i_1} = a_{i_2} = 0$. In this case, γ_{i_1} and γ_{i_2} are linearly dependent.

Case 2: There exists at most one vector whose first component is zero. Without loss of generality, assume $a_i \neq 0, i = 1, \cdots, k-1$. We have $\gamma_i = a_i(1, \frac{b_i}{a_i}), i = 1, \cdots, k-1$. Since $k \geq q + 2$, there exist $i_1 \neq i_2$ such that $\frac{b_{i_1}}{a_{i_1}} = \frac{b_{i_2}}{a_{i_2}}$. Thus γ_{i_1} and γ_{i_2} are linearly dependent.

In both cases, we can find two linearly dependent vectors. This contradiction yields the result. ∎

Theorem 6.9: The bound in Theorem 6.7 is tight.

Proof: We construct a minimal feasible region graph by adding some X_2 sink regions to Fig.14 (a), as follows.

1) For $j \in \{2, \cdots, n-1\}$, add an X_2 sink region as a common child of Q_j and the X_1 source region;
2) For Q_i and Q_j, which are not adjacent, add an X_2 sink region as a common child of Q_i and Q_j.

Denote the resulted region graph by $RG(D)$ and the corresponding region set by D. One can check that $RG(D)$ is still a minimal feasible region graph. We now prove that the field size for any linear code of $RG(D)$ is at least $\sqrt{2N - 7/4} + 1/2$.

Note that in Fig. 14 (a), there are $n-2$ coding regions not adjacent to Q_1, $n-3$ coding regions not adjacent to each $Q_j, j = 2, \cdots, n-1$, and $n-2$ coding regions not adjacent to Q_n. Thus, we have added $[2(n-2)+(n-2)(n-3)]/2 = (n^2 - 3n+2)/2$ sink regions in step 2) and the total number of sink regions of $RG(D)$ is $N = (n+2)+(n-2)+(n^2-3n+2)/2 = (n^2+n+2)/2$. So, we have $\sqrt{2N - 7/4} + 1/2 = n + 1$.

Second, if $\tilde{C} = \{d_R; R \in D\}$ is a linear code of $RG(D)$ over $\mathbb{F}$, then we declare that d_{Q_i} and d_{Q_j} are linear independent for any coding regions $Q_i \neq Q_j$. In fact, by the

construction of $RG(D)$, Q_i and Q_j have a common X_2 child. Thus, if d_{Q_i} and d_{Q_j} are linear dependent, then there exist $k_1, k_2 \in \mathbb{F}$ such that $k_1 \cdot d_{Q_i} = k_2 \cdot d_{Q_j} = (0, 1)$. Again by the construction of $RG(D)$, one can have $d_{Q_\ell} = k_\ell \cdot (0, 1)$ for each Q_ℓ downstream Q_i and/or Q_j, which is impossible because an X_1 sink region exists. By adding $(1, 0)$ and $(0, 1)$, we have totally $n + 2$ mutually linearly independent vectors in a solution. Note that $n + 2 \leq q + 1$ (Lemma 6.8) and we have $\sqrt{2N - 7/4} + 1/2 = n + 1 \leq q$. ■

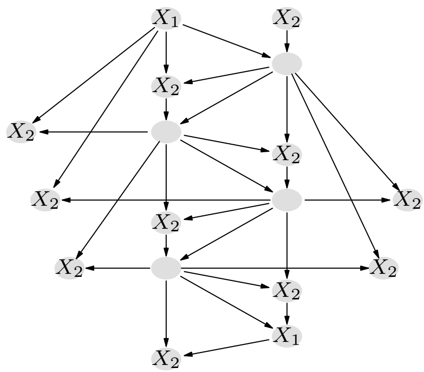


Fig 17. A minimal feasible region graph with $n = 4$ coding regions and $N = (n^2 + n + 2)/2 = 11$ sink regions.

Example 6.10: Fig.17 plots a region graph $RG(D)$ constructed as in the proof of Theorem 6.9. $RG(D)$ has $n = 4$ coding regions and $N = (n^2 + n + 2)/2 = 11$ sink regions. By Theorem 6.9, $\mathbb{F}_5$ ensure a linear solution of $RG(D)$.

VII. Conclusions and Discussions

We investigated the encoding complexity of the 2-SMNC problem by proposing a region decomposition method. It showed that when the network is decomposed into mutually disjointed regions, a network coding solution can be easily obtained from some simple labeling operations on the region graph and by decentralized assigning encoding kernels. All the processes of the region decomposition, the region labeling, and the code construction can be done in time $O|E|$.

We further reduced a feasible region graph into a minimal feasible one by deleting links and/or combining nodes of the region graph. It showed that the minimal feasible region graph have some marvelous local properties, from which we derived bounds on the encoding links and on the required field size.

There are some interesting issues need further investigate. For example, given a 2-SMNC network, one may get different minimal feasible region decompositions from different feasible region decompositions. We do not known if these minimal feasible region graphs are with a same topology but we did not ever find a contra example. Another valuable topic is how to apply the region decomposition method to an acyclic network with $n > 2$ multicast sessions. In this case, region decomposition can be performed similarly and the basic region decomposition can also be obtained. However, the labeling process becomes very complicated and new opinions need be introduced in order to get some valuable results. Beyond these issues, region decomposition on *cyclic* networks, vector linear codes for 2-SMNC problems are also interesting topics worthy of consideration.

Acknowledgment

This paper is supported by NSF of China (No. 10990011),the research fund for the Doctoral Program of Higher Education of China and by a grant from the University Grants Committee of the Hong Kong Special Administrative Region, China (Project No. AoE/E-02/08)

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204-1216, Jul. 2000.
[2] S.-Y. R. Li, R. W. Yeung, and N. Cai, "Linear network coding," *IEEE Trans. Inf. Theory*, vol. 49, no. 2, pp. 371-381, Feb.2003.
[3] C. K. Ngai and R.W.Yeung, "Multisource network coding with two sinks," in *Proc. Int. Conf. Communications, Circuits and Systems(IC- CCAS)*, vol. 1, Chengdu, China, Jun. 2004, pp. 34-37.
[4] C.-C. Wang and N.B. Shroff, "Intersession network coding for two simple multicast sessions," *The 45th Annual Allerton Conference on Communication, Control, and Computing (Allerton 2007),* Sept. 2007.
[5] C.-C. Wang and N. B. Shroff, "Pairwise Intersession Network Coding on Directed Networks," *IEEE Trans. Inf. Theory*, vol. 56, no. 8, pp. 3879-3900, Aug. 2010.
[6] N. Harvey, R. Kleinberg, and A. Lehman, "On the capacity of informa tion networks," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2345-2364, June 2006.
[7] G. Kramer and S. Savari, "Edge-cut bounds on network coding rates," *Journal of Network and Systems Management*, vol. 14, no. 1, pp. 49-67 March 2006.
[8] X. Yan, J. Yang, and Z. Zhang, "An outer bound for multisourc multi sink network coding with minimum cost consideration," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2373-2385, June 2006
[9] X. Yan, R. Yeung, and Z. Zhang, "The capacity region for multi-sourc multi-sink network coding," in *Proc. IEEE Int'l Symp. Inform. Theor*. Nice, France, 2007.
[10] D. Traskov, N. Ratnakar, D. Lun, R. Koetter, and M. Medard, "Network coding for multiple unicasts: An approach based on linear optimization," in *Proc. IEEE Int'l Symp. Inform. Theory*. Seattle, USA, July 2006.
[11] Y. Wu, "On constructive multi-source network coding," in *Proc. IEEE Int'l Symp. Inform. Theory*. Seattle, USA, July 2006, pp. 1349-1353.
[12] A. R. Lehman and E. Lehman, "Complexity classification of network information flow problems," in *Proc. 41st Annu. Allerton Conf. Communication Control and Computing*, Monticello, IL, Oct. 2003.
[13] R. Koetter and M. Mdard, "An algebraic approach to Network coding," *IEEE/ACM Trans. Netw*., vol. 11, no. 5, pp. 782-795, Oct. 2003.
[14] S. Jaggi, P. Sanders, P. A. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen, "Polynomial time algorithms for multicast network code construction," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 1973-1982, Jun. 2005.
[15] N. J. A. Harvey, D. R. Karger, and K. Murota, "Deterministic network coding by matrix completion," *Annual ACM-SIAM Symposium on Discrete Algorithm*, 2005.
[16] T. Ho, M. Medard, R. Kotter, D. R. Karger, M. Effros, J. Shi, and B. Leong, "A Random Linear Network Coding Approach to Multicast," *IEEE Trans. on Info. Theory*, vol. 52, no. 10, pp. 4413-4430, Oct 2006.
[17] C. Fragouli and E. Soljanin,"Information flow decomposition for network coding," *IEEE Trans. Inf. Theory*, vol. 52, no. 3, pp. 829-848, Mar. 2006.
[18] M. Langberg, A. Sprintson, and J. Bruck, "The encoding complexity of network coding," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2386-2397, June. 2006.
[19] R. Dougherty, C. Freiling, and K. Zegger, "Linearity and solbability in Multicast Networks," *IEEE Trans. Inf. Theory*, vol. 50, no. 10, pp. 2243-2256, Oct. 2004.
[20] R. Yeung, S.-Y. Li, and N. Cai, Network Coding Theory (Foundations and Trends in Communications and Information Theory). New York: Now, 2006.
[21] J. A. Bondy and U. S. R. Murty, Graph Theory with Applications. Amsterdam, The Netherlands: North-Holland, 1979.
[22] A. H. Ali, J.W. P. Hirschfeld, and H. Kaneta, "On the size of arcs in projective spaces," *IEEE Trans. Inf. Theory*, vol. 41, no. 5, pp. 1649-1656, Sep. 1995.